Neural Networks

March 21, 2025

Biological Underpinnings

The Biological Neuron

- Neurons are specialized cells that process and transmit information
- Structure includes:
  - **Soma** (cell body) - Contains the nucleus and processes inputs
  - **Dendrites** - Receive signals from other neurons
  - **Axon** - Transmits signals to other neurons
  - **Synapses** - Connection points between neurons
- The human brain contains over 80 billion neurons
- Each neuron may connect to thousands of others
- Signals can be excitatory (increase firing probability) or inhibitory (decrease firing probability)

From Biological to Artificial Neural Networks

Artificial Neuron

- Input connections receive signals
- Weighted sum of inputs
- Activation function (sigmoid, ReLU, etc.)
- Output connection transmits result
- Weighted connections
- Learning through backpropagation
- Parallel computing architectures

Key Simplifications in ANNs:

- Discrete time steps vs. continuous firing
- Simplified activation functions
- Uniform neuron types vs. diverse cell types
- Backpropagation vs. local learning rules

History

- **1943**: McCulloch & Pitts proposed the first mathematical model of a neuron
  - Binary threshold units performing logical operations
  - Demonstrated that networks of these neurons could compute any arithmetic or logical function
- **1949**: Donald Hebb published "The Organization of Behavior"
  - Introduced Hebbian learning: "Neurons that fire together, wire together"
  - First proposed learning rule for neural adaptation

The Perceptron Era (1950s-1960s)

- **1958**: Frank Rosenblatt introduced the Perceptron
  - First trainable neural network model
  - Binary classifier with adjustable weights
  - Could learn from examples using error-correction rule

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right)$$

$$\text{where } f(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

- **1969**: Minsky & Papert published "Perceptrons"
  - Proved fundamental limitations of single-layer perceptrons
  - Demonstrated they could not learn simple functions like XOR
  - Famous XOR problem became emblematic of perceptron limitations
- Impact:
  - Shift of focus to symbolic AI approaches
  - Need for multiple layers to solve non-linearly separable problems
  - Lack of effective training methods for multi-layer networks

- **1986**: Rumelhart, Hinton & Williams popularised backpropagation
  - Efficient algorithm for training multi-layer networks
  - Based on chain rule for computing gradients
  - Solved the XOR problem and more complex pattern recognition tasks

- Challenges that limited adoption:
  - Computational limitations (training was extremely slow)
  - Vanishing/exploding gradient problems in deep networks
  - Other approaches outperformed neural networks on many tasks
  - Need for large labelled datasets

Deep Learning Revolution

- **2006**: Hinton et al. introduced deep belief networks
  - Effective training of deep architectures
- **c. 2010**: GPU computing transformed neural network training
  - Orders of magnitude speedup for matrix operations
  - Enabled training of much larger networks
- **To present**: Many models, architectures and training approaches

Neuro-Evolution

Overview

- **Definition:** Neuroevolution is the application of evolutionary algorithms to optimise neural networks.
- Also adopted in Artificial Life as a means to explore different learning approaches.
- **History:**
  - 1980s: First attempts at evolving neural networks
  - 1990s: Evolution of fixed-topology networks
  - 2002: NEAT algorithm (Stanley & Miikkulainen)
  - 2009: HyperNEAT for large-scale networks
  - 2017 on: Deep neuroevolution
- **Main approaches:**
  - Direct encoding (weights, topologies)
  - Indirect encoding

Why Neuroevolution?

Benefits:

- Global optimisation (less prone to local optima)
- Can optimise both architecture and hyperparameters
- Useful approach when architecture is unknown
- Useful on highly multimodal landscapes

Neuroevolution in Artificial Life

- **Neural networks as “brains":**
  - Controllers for artificial organisms
  - Enable complex behaviors and adaptation
- **Biological inspiration:**
  - Evolution of nervous systems
  - Environmental pressures driving cognitive complexity
- **Goal:** Understand how intelligence emerges through evolutionary processes

Open-Ended Evolution

- **Definition:** Continuous adaptation and complexity growth
- **Challenges in ALife:**
  - Creating sufficient environmental complexity
  - Maintaining selective pressure over time
  - Avoiding evolutionary dead-ends
- **Neural networks in open-ended evolution:**
  - Increasing network complexity correlates with behavioral complexity
  - Incremental evolution builds on previous capabilities
- **Research frontier:** Creating truly open-ended neural evolution

Simple Neuroevolution

- **Fixed network topology:**
  - Architecture predetermined (e.g., layers, connectivity)
  - Only weights are evolved
- **Encoding strategies:**
  - Direct: Each weight is a separate gene
- **Genetic operators:**
  - Mutation: Random perturbations to weights
  - Crossover: Combining weights from parents
- **Advantages:** Simple, efficient
- **Limitations:** Architecture constraints

Neuro-evolution Process

1. Initialization: Generate initial population of neural networks.
2. Evaluation: Assess the fitness of each network on a task.
3. Selection: Choose networks to reproduce based on fitness.
4. Reproduction: Create new networks through crossover and mutation.
5. Repeat: Iterate through generations until convergence.

Representation

- Direct coding
- Marker based encoding
- Indirect coding

NEAT (NeuroEvolution of Augmenting Topologies)

- Simultaneous evolution of weights *and* topology
- Start with minimal network and grow complexity as needed
- Speciation to protect innovations

Genetic operators:

- Weight mutation
- Add connection
- Add node
- Crossover with history tracking

- Advantages:
  - Exploration of large search spaces
  - Adaptation to dynamic environments
  - Effective for complex problem domains
- Applications of Neuro-evolution:
  - Evolutionary Robotics
  - Game Playing Agents

Artificial life models

- In addition to application to practical optimisation problems, the neuro evolution model has been adopted in a range of artificial life models where one can explore the interplay between population based learning (genetic algorithm), life time learning (NNs), and other forms of learning. Has led to some interesting results

Artificial life models

- Signalling
- Language evolution
- Movement behaviours
- Flocking/clustering
- Means to explore the interplay between different learning types

Artificial life - types of learning

- Population based learning - (modelled with GAS)

Artificial life - types of learning

- Population based learning - (modelled with GAS)
- Life time learning - (modelled with NNs)

Artificial life - types of learning

- Population based learning - (modelled with GAS)
- Life time learning - (modelled with NNs)
- Cultural learning - (allows communication between agents)

Artificial life - Baldwin effect

- Consider a population of agents (represented by NNs) subject to evolutonary pressure (GAs)
- Many theories have been proposed to explain the evolution of traits in populations - Darwinian, Lamarckian etc
- The Baldwin Effect is a concept in evolutionary biology that suggests learned behaviors acquired by individuals during their lifetime can influence the direction of evolution.

- Baldwin Effect:
  - Learned behaviors initially arise through individual learning and are not genetically encoded.
  - Over time, individuals with adaptive learned behaviors may have higher fitness, leading to differential reproduction.
  - Selection pressure favours those individuals with certain learned behaviors
  - Eventually, these once-learned behaviors may become innate or genetically predisposed in subsequent generations.
- Hinton and Nowlan experiments showing this effect

- Combining life time and evolutionary learning
  - Can evolve greater plasticity in populations -can evolve the ability to learn useful functions
  - Can be useful in changing environments
  - Allows populations to adapt
  - Examples in game play (cards, connect 4) and simulated robotics

Cultural leaning

- Allows agents to learn from each other
  - Shown to allow even greater plasticity in populations
  - Has been used in conjunction with life-team learning and population based learning
  - Has been used to model the emergence of signals, 'language', dialects

Summary

- Neural networks as learning approaches to a range of tasks (e.g classification, generation) has been successful in a range of domains (e.g. image, speech, text)
- Neuro-evolution has been shown to be useful in a range of domains
- Neural nets provide a nice model in artificial life systems

Case Study: Evolved Communication

- **Multi-agent communication:**
  - Agents with neural signaling networks
  - No pre-defined communication protocols
  - Must evolve signals and interpretations
- **Key findings:**
  - Communication emerges when beneficial
  - Signal complexity matches task complexity
- **Applications:**
  - Origin of language models
  - Emergent semantics
  - Multi-agent coordination

Case Study: Predator-Prey Coevolution

Experimental setup:

- Populations of predator and prey agents
- Neural controllers for sensing and movement
- Evolving in shared environment

Red Queen dynamics:

- Continuous arms race
- Adaptation and counter-adaptation
- No stable equilibrium

Evolving Deep Neural Networks

Challenges:

- High-dimensional search spaces
- Computational requirements
- Efficient encoding of complex architectures

Conclusion

- **Summary:**
  - Neuroevolution is an interesting alternative gradient-based methods
  - In ALife, provides models of cognitive evolution
  - Many approaches: direct, indirect